E-commerce site

- By Hossam Mohamed (ha223si)
- Mariya Al Nasrala (ma227ji)

Conceptual Classes and Attributes (Analysis)

1. Customer

- ***Attributes***: customerID, username, password, email, phoneNumber, address, paymentInfo (e.g., card details), orderHistory (list of Orders) & wishlist (list of Products).
- ***Explanation***: Customers are the primary users of the system, purchasing products and managing their accounts. Their personal details, payment methods, and order history are key for e-commerce operations like account management, order tracking, and recommendations.

2. Product

- ***Attributes***: productID, name, description, category (e.g., laptops, accessories, components), brand, price, stockQuantity, warrantyPeriod, compatibility (optional, e.g., CPU socket type for a motherboard), rating (average customer rating) & images (list of URLs to images).
- ***Explanation***: The product class represents the tech items sold on the site. Each product has attributes relevant to e-commerce, such as price, category, and stock. In the computer products domain, compatibility (e.g., between a CPU and motherboard) and warranty period are unique attributes to capture.

3. Order

- ***Attributes***: orderID, customerID (reference to Customer), orderDate, products (list of product references), totalAmount, shippingStatus (e.g., pending, shipped, delivered), paymentStatus (paid/unpaid), paymentMethod (e.g., credit card, PayPal) & shippingAddress.
- ***Explanation***: The order class tracks each purchase made by a customer. It records essential information like which products were bought, the payment method, and the shipping status.

4. Shopping cart

- ***Attributes***: cartID, customerID (reference to Customer), products (list of Product references) & totalAmount.
- ***Explanation***: The shopping cart allows customers to temporarily store products before purchasing them. It holds a list of selected products and dynamically calculates the total cost as items are added or removed.

5. Payment

- ***Attributes***: paymentID, orderID` (reference to Order), amount, paymentDate, paymentMethod` (e.g., credit card, PayPal, bank transfer) & transactionStatus (successful/failed/pending)
- ***Explanation***: Payment processing is crucial in e-commerce. The payment class records the details of each transaction, including the method and status. Third-party systems like PayPal or credit card processing services would interact with this class.

6. Admin
Attributes: adminID, username & password
Explanation: Administrators are responsible for managing the platform. They add or modify products, track orders, and update stock levels. They also monitor payment statuses and customer service interactions.

7. Review
- ***Attributes***: reviewID, productID (reference to Product), customerID (reference to Customer), rating (numeric rating, 1–5 stars), comment & reviewDate
- ***Explanation***: The review class allows customers to provide feedback on purchased products. This feedback is important for other potential buyers and helps improve product listings. The system can aggregate ratings to provide an average score for each product.

8. Category
- ***Attributes***: categoryID, name (e.g., laptops, desktops, accessories) & description
- ***Explanation***: The category organizes products into logical groups. Each product is assigned to one or more categories, helping customers filter and search through the available items efficiently.

9. Warranty
- ***Attributes***: warrantyID, productID (reference to Product), customerID` (reference to Customer), warrantyStartDate, warrantyEndDate & terms (e.g., what is covered)
- ***Explanation***: The warranty class tracks warranty coverage for purchased products. For electronics, warranties are crucial, and customers need an easy way to check warranty periods and coverage in case of damage or malfunction.

Relationships/Associations (Analysis)

1. Customer to Order (1-to-many)
A customer can place multiple orders, but each order belongs to a single customer. This relationship allows tracking the complete order history for every customer.

2. Customer to Review (1-to-many)
A customer can write multiple reviews, but each review is linked to a single product. This relationship is important for feedback and ratings, which can influence other shoppers.

3. Product to Order (many-to-many)
An order can contain multiple products, and a product can be included in many orders. This many-to-many relationship is essential for representing real-world purchases.

4. Product to Category (many-to-many)
- A product can belong to multiple categories, and each category can contain multiple products. This relationship helps with product browsing and filtering on the website.

5. Product to Review (1-to-many)
- Each product can have multiple reviews from different customers, but each review is tied to a single product. This relationship helps in aggregating customer feedback.

6. Product to Warranty (1-to-1)

- Each product sold comes with its unique warranty for a specific customer. This one-to-one relationship tracks warranty details such as coverage period and terms for each sold product.

7. Customer to Shopping Cart (1-to-1)

- Each customer has only one active shopping cart at any time, but a shopping cart belongs to one customer. This relationship allows a customer to gather products they want to buy before finalizing the purchase.

8. Order to Payment (1-to-1)

- Each order must have one payment, and each payment corresponds to only one order. This one-to-one relationship ensures proper tracking of financial transactions.

Explanation of Concepts and Associations (Analysis)

1. The core actor in the system. ***Customers*** manage their accounts, purchase products, and interact with the store through reviews and wish lists. Their orders and payments are tracked to ensure a smooth shopping experience.

2. The main entity being sold. ***Products*** have unique characteristics like price, warranty, and compatibility details (for computer-related items). The system must track stock, ratings, and compatibility details to ensure the right information is presented to customers.

3. Orders represent a purchase event where a customer selects products and completes payment. Orders are central to e-commerce and require tracking across various stages (pending, shipped, delivered).

4. The shopping cart serves as a temporary container where customers collect products they plan to purchase. It calculates the total amount and helps transition the customer to checkout.

5. Payments must be processed securely, and tracking the status of each payment is critical to ensure no unpaid orders go through. The system should interact with third-party services for credit card or online payments.

6. Customer feedback is important in helping others make informed decisions. Each product needs ***reviews*** to increase credibility and improve the shopping experience.

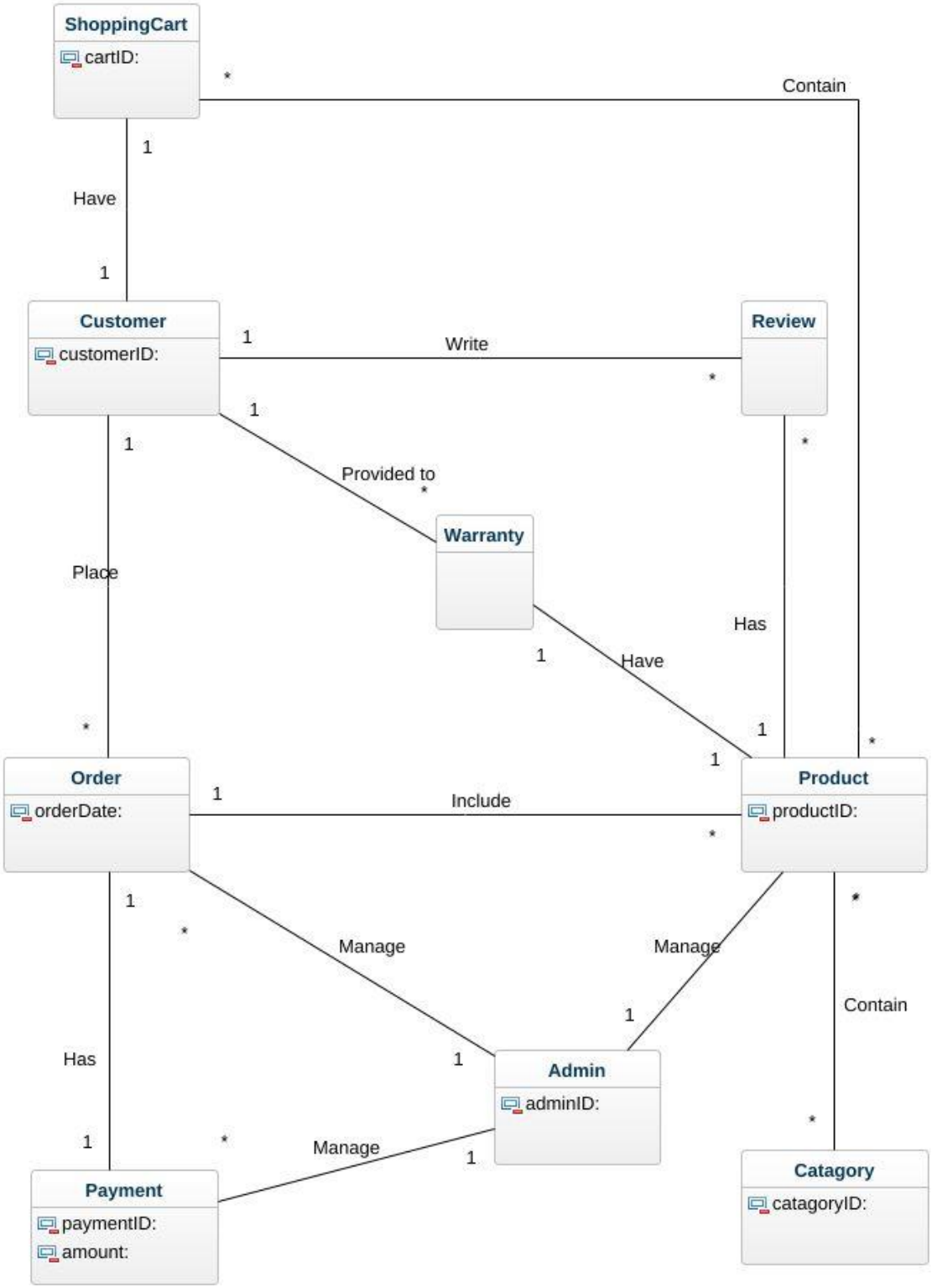
7. Unique to tech products, ***warranties*** offer protection to customers in case of defective products. The system must track warranty periods and allow customers to view their coverage easily.

8. Categories help organize products logically, allowing customers to filter their searches based on product type, brand, or other attributes. This improves the overall user experience when navigating through a large product catalogue.

Domain Model (Design)

Domain model summary

- ***Customer***: Registers, places orders, writes reviews, and manages their account.
- ***Product***: Sold by the store, categorized, reviewed, and tracked for compatibility and warranty.
- ***Order***: Records the purchase, tracks the products in the order, and links to the customer and payment.
- ***Payment***: Handles financial transactions for each order.
- ***Review***: Allows customers to rate and comment on purchased products.
- ***Category***: Organizes products into logical groups.
- ***Warranty***: Provides post-purchase product protection for tech products.


ShoppingCart
cartID:
Contain
Have
Customer
customerID:
Write
Review
Provided to
Warranty
Place
Has
Have
Order
orderDate:
Include
Product
productID:
Manage
Manage
Has
Admin
adminID:
Contain
Manage
Payment
paymentID:
amount:
Catagory
catagoryID: